TP n^{o}1 : initiation à MATLAB

Le but de ce premier TP est de (re)découvrir l'outil MATLAB à travers différents petits problèmes mathématiques avec la perspective d'en faire un rapport sous LaTeX.

1 La suite des grêlons

On considère la suite récurrente (u_n) définie par :

$$u_0 \in \mathbb{N}, \qquad \text{puis pour } n \in \mathbb{N} : \qquad u_{n+1} = \begin{cases} 3u_n + 1, & \text{si } u_n \text{ est impair,} \\ \dfrac{u_n}{2}, & \text{sinon.} \end{cases}$$

1. Créer une fonction `grelons.m` (avec des arguments d'entrée et de sortie pertinents), qui permette de simuler les premiers termes de la suite (jusqu'à un ordre souhaité).
2. Tester la fonction pour différentes initialisations. Que remarque-t-on ?
3. Créer un script `test_grelons.m` afin de conjecturer le comportement de la suite à l'infini. On cherchera à mettre en valeur le maximum de la suite, ainsi que le nombre d'itérations nécessaire pour atteindre le cycle final.
4. Modifier le script afin de le rendre fonctionnel pour un utilisateur. L'interface comprendra en particulier l'entrée par l'utilisateur de la condition initiale, l'affichage de résultats pertinents, ainsi que la possibilité de recommencer le processus avec une nouvelle donnée initiale sans avoir à relancer le programme.

2 Résolution de l'équation $f(x) = 0$

Soit f une fonction continue sur l'intervalle $[a, b]$. On suppose que f s'annule sur cet intervalle et on cherche à approcher numériquement le (ou les) zéro(s) de cette fonction. On propose pour cela deux algorithmes, dont on fera une comparaison des performances.

2.1 Dichotomie

La méthode de dichotomie permet d'**approcher le zéro d'une fonction continue en cas de changement de signe**. Le principe est de tester le signe de la fonction au point courant, en divisant la taille de l'intervalle d'étude par deux à chaque itération (voir **Algorithme 1**). Cette méthode est basée sur le **principe des valeurs intermédiaires**.

1. Implémenter cet algorithme sous MATLAB.
2. Modifier le critère d'arrêt pour atteindre une tolérance ε sur le résidu.
3. Reprendre la question précédente avec un critère d'arrêt sur l'incrément.
4. Comparer les résultats et expliquer.

Algorithme 1 Algorithme de dichotomie

Paramètres d'entrées : f, a, b
- f fonction continue sur un intervalle I
- deux réels a et b de I tels que $f(a)f(b) < 0$

Paramètres de sorties : x, n
- x approximation du zéro de la fonction f
- n nombre d'itérations nécessaires pour atteindre x

$a_0 \leftarrow a$, $b_0 \leftarrow b$
$x_0 \leftarrow \dfrac{a+b}{2}$
Tant que $(x_n > a_n \ \&\ x_n < b_n)$ **Faire**
  Si $f(a_n)f(x_n) > 0$ **alors**
    $a_{n+1} \leftarrow x_n$, $b_{n+1} \leftarrow b_n$
    $x_{n+1} \leftarrow \dfrac{x_n + b_n}{2}$
  Sinon
    $a_{n+1} \leftarrow a_n$, $b_{n+1} \leftarrow x_n$
    $x_{n+1} \leftarrow \dfrac{x_n + a_n}{2}$
  Fin Si
Fin Tant que

2.2 La méthode de Newton

La méthode de Newton est une technique d'approximation des zéros d'une fonction dérivable dont la dérivée ne s'annule dans le voisinage du zéro. Elle est dans la plupart des cas d'ordre 2, ce qui en fait son succès auprès des numériciens.

On se donne une fonction f dérivable sur l'intervalle $[a, b]$. Pour résoudre numériquement l'équation $f(x) = 0$ dont on sait que la solution $\bar{x}$ vérifie $f'(\bar{x}) \neq 0$, la méthode de Newton consiste à générer la suite récurrente suivante :

$$\boxed{x_0 \text{ donné} \ ; \ \forall\, n \in \mathbb{N}, \ x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.}$$

1. Implémenter la méthode.
2. Appliquer la méthode à la fonction arctan en testant différentes initialisations. Que remarque-t-on ?
3. En étudiant la suite récurrente associée, justifier cette remarque. On donnera en particulier une approximation du voisinage dans lequel on converge nécessairement.
4. Proposer une solution pour assurer la convergence de la suite.
5. Comparer les deux méthodes proposées sur les cas-tests de votre choix.

3 Un peu de LaTeX pour finir

LaTeX est l'un des compagnons privilégiés de l'élève-ingénieur en Macs. La puissance de cet outil permet de rédiger des rapports de grande qualité esthétique autant que mathématique, les deux n'étant pas incompatibles, ainsi que de créer les présentations scientifiques les plus complexes.

En vous inspirant du fichier `TP1.tex` fourni, rédiger un rapport reprenant les différentes réponses aux questions du TP et en les illustrant avec des figures.

Les codes ainsi que le rapport sont à envoyer (compactés bien sûr) à l'adresse `yohan.penel@cea.fr` avant le **dimanche 20 décembre 2009**, **23h59**.